The paragraph starting on page 5 line 23 and continuing to page 6 line 20 has been amended to correct a spelling error and to correct references to ".SUBCKT" and ".ENDS".

Claim Rejections

The Examiner rejected claims 8, 9, 15, 16, 24, and 25 under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim subject matter which the applicant regards as the invention. The Examiner stated that pursuant to claims 8, 9, 15, 16, 24, and 25, it is unclear whether the "SUBCKT" should be preceded by a period as in claims 3, 10, and 19.

Claims 1, 3-9, 10, 12-16, 17, and 19-25 were rejected under 35 U.S.C. 112, second paragraph, as being incomplete for omitting essential structural cooperative relationships of elements, such omissions amounting to a gap between the necessary structural connections.

Claims 1, 10, and 17 have been amended to recite the terms: ".SUBCKT", ".ENDS", node name, output signal, and discrete circuit element description as part of the converted (Verilog to SPICE ) block design. This now provides the structural relationship between these claims and depending claims. Also, claims 1, 10 and 17 have been amended to recite a subcircuit name identified by a ".SUBCKT" heading such that references to ".SUBCKT" in depending claims refer to the heading and "subcircuit name" has replaced the previously used "SUBCKT name" in depending claims for clarity.

Claims 3-9 have been amended to recite the converted first design block to provide the structural relationship between the recited elements and claim 1 from which these claims depend.

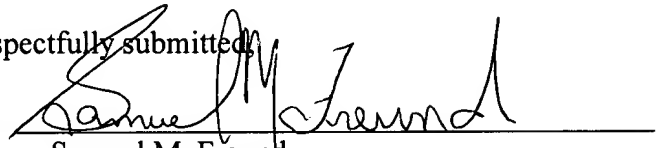
Claims 12-16 have been amended to correct references made to SUBCKT name and to SPICE, and to reference the SPICE file of claim 10 to provide the structural relationship of where recited elements exist.

Claims 19-25 have been amended to correct references made to SUBCKT name and to SPICE, and to reference the converted first design block of claim 17 to provide the structural relationship of where recited elements exist.

In view of the above amendments, this application is now considered to be in condition for allowance and such action as earnestly solicited.

Dated this 2nd day of April 2003.

Respectfully submitted,

By: 

Samuel M. Freund
Attorney for Applicant
Reg. No. 30,459
The Law Offices of William W. Cochran, LLC
3555 Stanford Road, Suite 230
Fort Collins, CO 80525
Phone: (970) 377-6363
Fax: (970) 207-1985

Markup copy of the paragraph starting on page 5 line 23 and continuing to page 6 line 20.

Figure 5 is a flowchart of a SPICE to Verilog translation process. The translation process 500 begins with SPICE netlist 502 that is opened and read in at step 504. Table 1 below lists an example SPICE file input and Table 2 below lists an example Verilog file output. At step 506, instantiations of the heading ["SUBCKT"] ".SUBCKT" are translated to "module". A circuit name, and input and output signal names, may follow the ["SUBCKT"] ".SUBCKT" heading. The circuit name may be identified as the first [work] word following ["SUBCKT"] ".SUBCKT" and may be utilized as the module name. Input and output signal names follow the circuit name and may be may be employed in the Verilog output file. For example, the ["SUBCKT] ".SUBCKT INV_CHAIN Z A" line of table 1 may be employed to generate the "module INV_CHAIN (Z, A)" entry listed in table 2. Retaining the circuit name and input and output signal names allows simplified association of the SPICE input file with the Verilog output file. Alternately, new circuit and signal names may be introduced in the translation process. At step 508, ["ENDS"] ".ENDS" statements are translated to "endmodule". At step 510, "x" elements are translated to Verilog format. The 'x' elements may comprise an instance name, signal nodes, and a circuit element descriptor. For example, the "XINV3 2 3 INV" line listed in table 1 may be translated to the "INV INV3 (.Z(3), .A(2)):" line listed in table 2. The translation of instance names ("XINV1" to "INV1" in the above example), may employ simple truncation of a leading "x" in element names, or may employ renaming. Renaming may be total or in part and may employ a lookup table, an algorithm, or rules to define the instance name. Advantageously, the present invention allows the hierarchy of the SPICE design and signal name associations to be maintained. The signal node names of the SPICE file may be employed to define wires in Verilog, and to specify signal connections to Verilog circuit elements. As may be observed in table 2, the signal nodes listed in table 1 are defined as wires. The circuit element descriptor may be translated to Verilog syntax. [IN] In the examples of tables 1 and 2, INV is used for both SPICE and Verilog syntax. At step 512, discrete circuit elements, such as resistors, capacitors, transistors, and inductors, for example, are

removed. At step 514, the modified file is written out to produce Verilog structured netlist 516.

Markup Copy of Amended Claims

[We claim:] What is claimed is:

1. (Amended) A method for designing an integrated circuit comprising:

partitioning a design into a plurality of function blocks;

designing a first one of said plurality of function blocks employing Verilog to produce a first block design;

designing a second one of said plurality of function blocks employing SPICE to produce a second block design;

converting said first block design from Verilog to SPICE to produce a converted first block design comprising a file including a subcircuit name identified by ".SUBCKT" heading, at least one node name, at least one discrete circuit element description, at least one output signal name, and a ".ENDS" statement;

simulating operation of said converted first block design and said second block design; and

translating said converted first block design from SPICE to Verilog to produce a translated first block design.

3. (Amended) The method of claim 1 wherein said step of translating further comprises:

changing said ".SUBCKT" [instances] heading in said converted first block design to "module".

4. (Amended) The method of claim 1 wherein said step of translating further comprises:

changing said ".ENDS" [statements] statement in said converted first block design netlist to "endmodule".

5. (Amended) The method of claim 1 wherein said step of translating further comprises:

deleting [a] said at least one discrete circuit element description in said converted first block design.

6. (Amended) The method of claim 1 wherein said step of translating further comprises:

defining a wire name corresponding to [a SPICE] said at least one node name in said converted first block design.

7. (Amended) The method of claim 1 wherein said step of translating further comprises:

identifying [a SPICE] said at least one output signal name in said converted first block design and defining a Verilog output signal using said output signal name.

8. (Amended) The method of claim 1 wherein said step of translating further comprises:

identifying [a] said [SUBCKT] subcircuit name[;] .

9. (Amended) The method of claim 8 wherein said step of translating further comprises:

employing said [SUBCKT] subcircuit name as a module name[;] .

10. (Amended) A method for translating a SPICE netlist to Verilog comprising:

opening a SPICE file comprising a subcircuit name identified by a ".SUBCKT" heading, at least one input signal name, at least one circuit element, at least one discrete circuit element description, an output signal name, and a .ENDS statement;

translating said ".SUBCKT" [instantiations] heading in said SPICE file to "module";

translating said ".ENDS" [statements] statement in said SPICE file to "endmodule";

translating [SPICE circuit elements] said at least one circuit element in said SPICE file to Verilog format; and

removing said at least one discrete circuit [elements] element description.

12. (Amended) The method of claim 10 wherein said step of translating further comprises:

identifying [a SPICE] said input signal name in said SPICE file and defining a Verilog wire employing said input signal name.

13. (Amended) The method of claim [10] 12 wherein said step of [translating] identifying further comprises:

identifying [an] said input signal name through a naming convention.

14. (Amended) The method of claim [10] 12 wherein said step of [translating] identifying further comprises:

identifying [an] said input signal name through a predefined delimiter.

15. (Amended) The method of claim 10 further comprising:

identifying [a SUBCKT] said subcircuit name in said SPICE file.

16. (Amended) The method of claim 15 wherein said step of translating further comprises:

employing said [SUBCKT] subcircuit name as a Verilog module name.

17. (Amended) An integrated circuit produced by the steps of:

partitioning a design into a plurality of function blocks;

designing a first one of said plurality of function blocks employing Verilog to produce a first block design;

designing a second one of said plurality of function blocks employing SPICE to produce a second block design;

converting said first block design from Verilog to SPICE to produce a converted first block design comprising a file including a subcircuit name identified by a ".SUBCKT" heading, at least one node name, at least one discrete circuit element description, at least one output signal name, and a ".ENDS" statement;

simulating operation of said converted first block design and said second block design; and

translating said converted first block design to Verilog to produce a translated first block design.

19. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

changing said ".SUBCKT" [instances] heading in said converted first block design to "module".

20. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

changing said ".ENDS" [statements] statement in said converted first block design to "endmodule".

21. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

deleting [a] said discrete circuit element description in said converted first block design.

22. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

defining a Verilog wire name corresponding to [a SPICE] said at least one node name in said converted first block design.

23. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

identifying [a SPICE] said at least one output signal name in said converted first block design and defining a Verilog output signal using said output signal name.

24. (Amended) The integrated circuit of claim 17 wherein said step of translating further comprises:

identifying [a SUBCKT] said subcircuit name.

25. (Amended) The integrated circuit of claim 24 wherein said step of translating further comprises:

employing said [SUBCKT] subcircuit name as a module name.